

Spot Duplicate Concepts

This document describes the domain model, highlights duplicate concepts, and outlines decisions to unify or separate them.

1. Overview

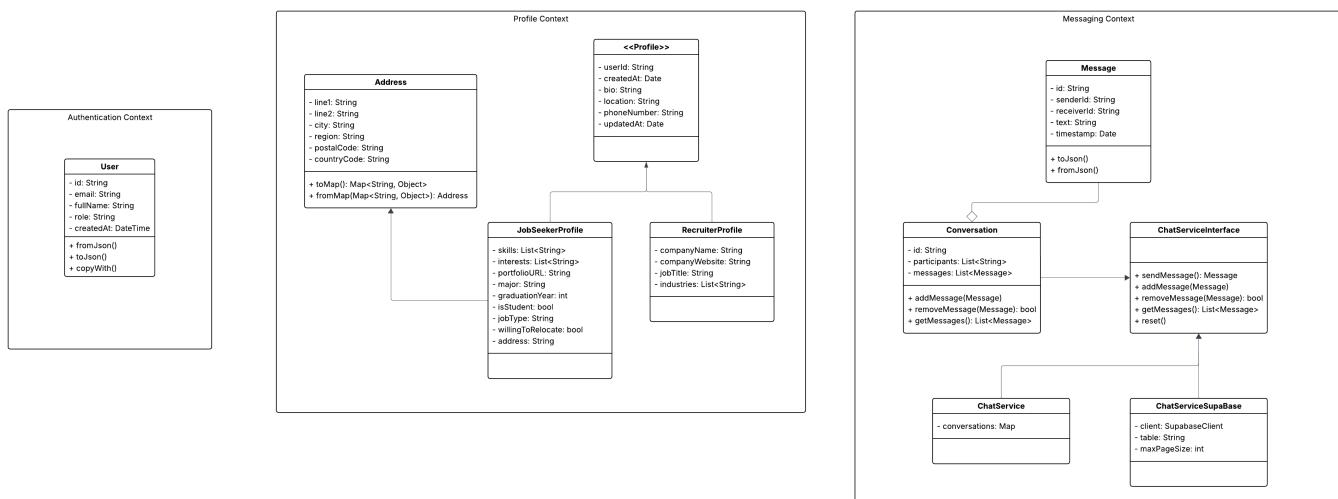
This document details:

- The bounded contexts in the system
- The main classes and their attributes/methods
- Duplicates identified and refactoring decisions
- A diagram to visually represent the domain model

2. Domain Model UML Diagram

https://lucid.app/lucidchart/1ccab065-589b-40c7-aca7-4785e74eba0e/edit?view_items=y9f.1GEqSfm_%2CeRh.OWc.GM27%2C1bg.0UmO4g1v%2CBVh.c2iFXcir%2CW0h.pmzy0qMm%2Cvli.iAZf8oMP%2Czli.tIKsxTJi%2Cpxh.6IVve.hr%2CG8f.SnYCNWr%2Ch9f.ldz3FEzu%2Cpdg.pT_.wDbf%2CQDh.5xTB_DLd%2C1mf.yjueAYCa%2CKdg.-Zu718yw%2CjDh.VNzbE243%2CvXh.4.CgeXI%2CUvi.tYm6VvmC%2CUji.RDoNNmgk%2CPl.2wwlh6qF%2CULi.-qvxArwY&page=0_0&invitationId=inv_7464a615-dabb-411d-b640-66c5c92dea4e

or



3. Bounded Contexts

3.1 Authentication Context

Classes

- **User**

- Variables:
 - `id: String`
 - `email: String`
 - `fullName: String`
 - `role: String`
 - `createdAt: DateTime`
- Methods:
 - `fromJson()`
 - `toJson()`
 - `copyWith()`

Notes on duplicates

- None within this context.
-

3.2 Profile Context

Classes

- **Address**
 - Variables:
 - `line1: String`
 - `line2: String`
 - `city: String`
 - `region: String`
 - `postalCode: String`
 - `countryCode: String`
 - Methods:
 - `toMap(): Map<String, Object>`
 - `fromMap(Map<String, Object>): Address`
- **Profile** (abstract/generalized profile)
 - Variables:
 - `userId: String`
 - `createdAt: Date`
 - `bio: String`
 - `location: String`
 - `phoneNumber: String`
 - `updatedAt: Date`
 - Methods:
 - None
- **JobSeekerProfile**
 - Variables:
 - `skills: List<String>`
 - `interests: List<String>`

- portfolioURL: String
 - major: String
 - graduationYear: int
 - isStudent: bool
 - jobType: String
 - willingToRelocate: bool
 - address: String
- Methods:
 - None

- **RecruiterProfile**

- Variables:
 - companyName: String
 - companyWebsite: String
 - jobTitle: String
 - industries: List<String>
- Methods:
 - None

Notes on duplicates

- `StudentProfile` merged into `JobSeekerProfile` to reduce duplication.
-

3.3 Messaging Context

Classes

- **Message**

- Variables:
 - id: String
 - senderId: String
 - receiverId: String
 - text: String
 - timestamp: Date
- Methods:
 - toJson()
 - fromJson()

- **Conversation**

- Variables:
 - id: String
 - participants: List<String>
 - messages: List<Message>
- Methods:
 - addMessage(Message)
 - removeMessage(Message): bool

- `getMessages(): List<Message>`

- **ChatServiceInterface**

- Variables:
 - None
- Methods:
 - `sendMessage(): Message`
 - `addMessage(Message)`
 - `removeMessage(Message): bool`
 - `getMessages(): List<Message>`
 - `reset()`

- **ChatService**

- Variables:
 - `conversations: Map`
- Methods:
 - None

- **ChatServiceSupabase**

- Variables:
 - `client: SupabaseClient`
 - `table: String`
 - `maxPageSize: int`
- Methods:
 - None

Notes on duplicates

- `Message` class used consistently across all chat services.
 - `Conversation` unifies message storage; no duplicates present.
-

4. Duplicate Concepts Identified and Refactored

Original Concept	Duplicate / Similar	Decision
StudentProfile	JobSeekerProfile	Merge into JobSeekerProfile
Profile data in multiple classes	JobSeekerProfile & RecruiterProfile	Create an abstract class that JobSeekerProfile & RecruiterProfile can inherit/implement
Message class in ChatService and Supabase	Message	Unified under Messaging Context

5. Summary

- All redundant classes have been merged or unified.

- Shared Value Objects are reused across contexts.
 - The domain model is clearer and avoids duplicated structures.
-